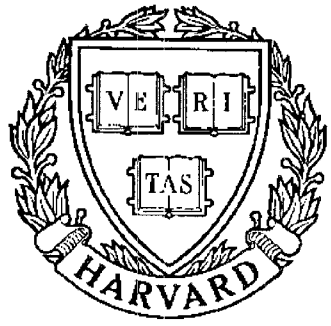
TECHNICAL RESEARCH REPORT

SYSTEMS RESEARCH CENTER



Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), Industry and the University

High Order Integration of Smooth Dynamical Systems: Theory and Numerical Experiments

by M. Austin

TR 91-98

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
1991		**00-00-1991 to 00-00-1991**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
High Order Integration of Smooth Dynamical Systems: Theory and Numerical Experiments	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Maryland,Systems Research Center,College Park,MD,20742**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**43**	

High Order Integration of Smooth Dynamical Systems : Theory and Numerical Experiments[1]

By: Mark Austin

Department of Civil Engineering and Systems Research Center,
University of Maryland,
College Park, MD 20742.

SUMMARY

This paper describes a new class of algorithms for integrating linear second order equations, and those containing smooth nonlinearities. The algorithms are based on a combination of ideas from standard Newmark integration methods, and extrapolation techniques. For the algorithm to work, the underlying Newmark method must be stable, second order accurate, and produce asymptotic error expansions for response quantities containing only even ordered terms. It is proved that setting the Newmark parameter γ to $1/2$ gives a desirable asymptotic expansion, irrespective of the setting for β. Numerical experiments are conducted for two linear and two nonlinear applications.

Keywords: Numerical Analysis, Extrapolation, Dynamics, Newmark Integration

[1] Also Submitted to International Journal of Numerical Methods in Engineering, November, 1991.

TABLE OF CONTENTS

1 INTRODUCTION

Engineers need fast and accurate computational methods for the dynamical design, analysis, and control of mechanical and structural systems. In the design and performance evaluation of dualspin and flexible satellites, for example, very accurate computational methods are needed for the long term prediction of position and attitude. Similarly, fast computational methods are needed to simulate and control flexible robotic manipulators and free flying robots. Since closed form analytic solutions are unavailable except for a few specific cases, discrete numerical integrators must be relied on to predict system behavior. Early numerical experiments, including ours [4,6], exposed weaknesses in the use of traditional numerical integrators. The use of standard off-the-shelf Runga-Kutta algorithms to integrate the dynamics of rigid body satellites spinning freely in space nearly always results in a `steady accumulation of energy`. Thus, they are a poor guide to the reliable prediction of long term dynamical phenomena, and highlight the need to design new numerical integrators to calculate time-dependent system responses.

To provide guidance in the formulation of these computational models, researchers are attempting to identify and classify the natural algebraic and geometric structures that these dynamical systems possess [15,17]. If energy and/or norms of angular momentum are are conserved in the continuous system response, then it is very desirable the same entities remain invariant in the discrete approximation. In the case of canonical Hamiltonian systems, Feng Kang [14] has shown that it is possible to construct symplectic schemes of arbitrary order accuracy, which are A-stable and conserve all invariants up to second order. Some recent algorithms [1,22] for noncanonical Hamiltonian systems have been shown to conserve energy exactly, but are only second order exact in their response estimates. As a

result, when large time-steps are used in simulation studies of long-term dynamical behavior, displacements can be completely out of phase after only several hundreds timesteps; examples may be found in Austin, Krishnaprasad, and Wang [1], and Hoff and Taylor [11].

Experience indicates that while numerical algorithms may be designed to conserve some integrals of motion, they cannot be expected (in general) to conserve all integrals of motion. Marsden et al. [17] suggest that numerical algorithms be crafted to conserve exactly some integrals of motion important for design, with other integrals possibly being conserved. This approach has several problems. First, it is not hard to find engineering applications where the equations of motion are so nonlinear, it is unlikely that a discrete approximation will be found to satisfy even one invariant of motion. And when the system is relatively uncomplicated, and such a discrete approximation can be found, performance may still be less than satisfactory (as already noted above). Moreover, it may be argued that even if a discrete numerical approximation theoretically conserves selected invariants exactly, its implementation will at most conserve the same quantities to machine precision; this will be approximately 16 decimal places accuracy for double precision calculations on standard engineering workstations.

A second strategy for developing numerical algorithms, and the approach that is followed in this paper, is to find efficient ways of systematically computing displacements and velocities to an arbitrarily high order of accuracy. This approach offers several computational benefits. First, it is easy to show that `all system invariants`, which are simple polynomial functions of displacement and velocity, will be preserved to the same order of accuracy as the displacements and velocities. Second, the equations of motion may be naturally partitioned for concurrent computations in a network of engineering workstations;

for details, the interested reader is referred to Austin and Voon [2].

The purpose of this paper is to present the formulation of a new algorithm that uses ideas from Newmark integration and extrapolation techniques to accurately compute long term dynamics. The scope of applications is limited to two classes of finite dimensional systems, namely:

[a] `Mass-Spring Systems` : Linear undamped (and damped) single (and multiple) degree of freedom systems moving under free and forced vibrations.

[b] `Connected rigid body assemblies` : Applications include the attitude dynamics of rigid body and dualspin spacecraft subject to zero external forces, dynamics of floating rigid body components connected by hinges, and the attitude control of free-flying robots. For discussions, see the work of Sreeneth [23,24], Yang [26], and Byrne [4]. Also, see Posbergh [19] for the formulation of a geometrically exact rod model as a series of springs and masses.

Numerical experiments are conducted to assess the effectiveness, and demonstrate the limitations of proposed algorithm.

2 EQUATIONS OF MOTION

We assume that the dynamical behavior of each application may be written as a family of n 2nd order differential equations:

$$M\left[\dot{x}(t), x(t)\right] \ddot{x}(t) + F\left[\dot{x}(t), x(t)\right] = P\left(t\right) \tag{1}$$

with initial conditions $x(0)$ and $\dot{x}(0)$. Here $x(t) = [x_1(t), x_2(t) \ \ldots \ x_n(t)]^T$ is a $(n \times 1)$ vector

of system displacements, $M[\dot{x}(t), x(t)]$ a $(n \times n)$ mass-type matrix, $F[\dot{x}(t), x(t)]$ a $(n \times 1)$ general force vector, and $P(t)$ a $(n \times 1)$ vector of external loads applied at the nodal degrees of freedom. An alternative, but equivalent form of (1) is obtained by setting $w(t) = \dot{x}(t)$, and rewriting the equations of equilibrium as $2n$ first order equations, i.e.:

$$\dot{y}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{w}(t) \end{bmatrix} = f[t, y(t)] = \begin{bmatrix} w(t) \\ M^{-1}[w(t), x(t)] \cdot [P(t) - F(w(t), x(t))] \end{bmatrix} \tag{2}$$

In the case of linear second order dynamical systems, for example, often $F(w(t), x(t)) = [c]w(t) + [k]x(t)$ - here $[c]$ and $[k]$ are $(n \times n)$ damping and stiffness matrices - and $M[\dot{x}(t), x(t)]$ is a constant mass matrix $[m]$. Equation (2) may be written:

$$[\dot{y}(t)] = \begin{bmatrix} \dot{x}(t) \\ \dot{w}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\ -[m]^{-1}[k] & -[m]^{-1}[c] \end{bmatrix} \cdot \begin{bmatrix} x(t) \\ w(t) \end{bmatrix} + \begin{bmatrix} 0 \\ [m]^{-1}[P(t)] \end{bmatrix} \tag{3}$$

Equation (3) underscores the fact that the complete dynamics are characterized by time variations in displacements and velocities alone. Accelerations are recovered via back substitutiion into (1). From a computational standpoint, it is desirable to work with (1) because it has only n equations, versus 2n for equations (2) and (3).

3 ONESTEP INTEGRATION SCHEMES

Suppose that solutions to (1)-(3) are required for time $t \epsilon [t_q, t_{q+\triangle t}]$. Irrespective of whether the underlying dynamics are described as families of first order or second order equations, numerical solutions require accurate approximations to:

$$y_k(t_{q+\triangle t}) = y_k(t_q) + \int_{t_q}^{t_{q+\triangle t}} f_k(\tau, y(\tau)) d\tau \tag{4}$$

where $f_k(t, y(t))$ is the $\mathtt{k}^{th}$ component of the equation of motion. In problem formulations based directly on d'Alembert's principle, $[y_k(t), f_k(t, y(t))]$ pairs exist for displacement and velocity, and velocity and acceleration, respectively. Similarly, in problem formulations based on Lagrange's equations and/or Hamilton's equations, $[y_k(t), f_k(t, y(t))]$ pairs exist for the generalized co-ordinates and momenta.

A wide variety of onestep numerical integration schemes approximate (4) by dividing the time interval $[t_q, t_{q+\triangle t}]$ into N equal intervals of distance $h = t_{q+h} - t_q$, and sequentially applying the `explicit` onestep formula

$$y(t_{q+h}) = y(t_q) + h \cdot \Phi(t_q, y(t_q), h) + O(h^{p+1}). \tag{5}$$

Here $\Phi : \mathbb{R} \times \mathbb{R}^{2n} \times \mathbb{R} \rightarrow \mathbb{R}^{2n}$ is called the increment function of the method, and $y(t_q)$ is the exact solution at time t_q. For numerical methods that have `implicit` increment functions, we write

$$y(t_{q+h}) = y(t_q) + h \cdot \Psi\left(t_q, t_{q+h}, y(t_q), y(t_{q+h}), h\right) + O(h^{p+1}) \tag{6}$$

where $\Psi : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^{2n} \times \mathbb{R}^{2n} \times \mathbb{R} \rightarrow \mathbb{R}^{2n}$. It is important to note that when the onestep truncation error is $O(h^{p+1})$, the global error is a combination of local plus transported errors, and is $O(h^p)$. A detailed proof is given in Chapter 3 of Hairer et al. [9]. Moreover, multistep methods may be interpreted in the framework of (4) by simply rewriting the numerical schemes as onestep methods in a higher dimensional space [9].

4 BACKGROUND TO EXTRAPOLATION METHODS

Perhaps the most straight forward way of increasing numerical accuracy is to divide the interval $[t_q, t_{q+\triangle t}]$ into smaller increments of h. While this strategy may be acceptable for short periods of simulation, it places the designer at odds with the need to compute the long-term time evolution of dynamical systems without resorting to excessive computational effort.

The algorithm developed in this paper hinges on the existence of an asymptotic expansion for the global error in the numerical method. Although the benefits of these expansions were known to Richardson [20,21], the formal theory for their existence is due to Gragg [8] and Stetter [25]. Gragg's ideas are most succinctly stated:

(a) **Theorem :** (due to Gragg) Suppose that a given method with sufficiently differentiable (smooth) increment function $\Phi(\cdot)$ satisfies the consistency condition $\Phi(t_q, y(t_q), 0) = f(t_q, y(t_q))$ and possesses an asymptotic expansion:

$$\begin{aligned} y(t_{q+h}) =& y(t_q) + h \cdot \Phi(t_q, y(t_q), h) + \\ & d_{p+1}(t)h^{p+1} + \cdots + d_{p+k}(t)h^{p+k} + O(h^{p+k+1}) \end{aligned} \tag{7}$$

for the local error. If N steps of integration are computed to cover the time interval $\triangle t$, (i.e. $\triangle t = t_{q+\triangle t} - t_q = N \cdot h$), then the global error has an asymptotic expansion of the form:

$$u(t_{q+\triangle t}) - y(t_{q+\triangle t}) = e_p(t) \cdot \triangle t^p + \cdots + e_{p+k}(t) \cdot \triangle t^{p+k} + E_h(t)\triangle t^{p+k+1} \tag{8}$$

where $u(t_{q+\triangle t})$ is the numerical solution at time $t_{q+\triangle t}$, and $e_p(t)$ are solutions to the

inhomogeneous equations:

$$\dot{e}_p(t) = \frac{df}{dy}(t, y(t))e_p(t) - d_{p+1}(t), \quad \text{such that } e_p(t_q) = 0 \tag{9}$$

and $E_h(t)$ is bounded for $t_q \leq t \leq t_{q+\Delta t}$ and $0 \leq h \leq h_o$. $f(\cdot)$ and $\Phi(\cdot)$ are as defined in equations (2) and (5), respectively.

An abbreviated version of the proof to Gragg's theorem may be found in Hairier and Lubich [11]. It is important to note that:

[a] The existence of equations (7)-(9) requires $f(\cdot)$ be sufficiently differentiable over the complete time interval of interest; hence use of the word `smooth` in the title of the paper. This condition automatically precludes the use of extrapolation methods for applications containing sharp material (or geometric) discontinuities.

[b] As stated, equations (7)-(9) are for an explicit increment function. Subsequent work by Setter [25] has established that asymptotic expansions of the form (8) also exist for implicit onestep methods.

[c] Extrapolation techniques require the underlying numerical method to be stable. Although explicit numerical integrators, such as Euler, are much less computational work that implicit schemes, they are notorious for being only conditionally stable [7,9]. In this study, base steps are computed with implicit increment functions that are A-stable.

[d] Preservation of the asymptotic expansion requires that the equations (7) be `solved exactly` at each timestep. However, it is demonstrated in the following sections that even when nonlinearities force the equations to be solved iteratively, very favorable increases in numerical accuracy are still possible.

The most important extrapolation methods are those that remain invariant under the switching of parameters $t_q \leftrightarrow t_{q+h}$, $y(t_q) \leftrightarrow y(t_{q+h})$, and integration order $h \leftrightarrow -h$. For example, if:

$$y(t_{q+h}) = y(t_q) + h \cdot \Psi(t_q, t_{q+h}, y(t_q), y(t_{q+h}), h) + O(h^3) \tag{10}$$

is equivalent to

$$y(t_q) = y(t_{q+h}) - h \cdot \Psi(t_{q+h}, t_q, y(t_{q+h}), y(t_q), -h) + O(h^3) \tag{11}$$

then the method is said to be symmetric. Instead of having an asymptotic expansion of the form:

$$\begin{aligned} u(t_{q+\triangle t}) =& y(t_{q+\triangle t}) + A_2 \triangle t^2 + A_3 \triangle t^3 + A_4 \triangle t^4 + \\ & A_5 \triangle t^5 + A_6 \triangle t^6 + A_7 \triangle t^7 + \cdots \end{aligned} \tag{12}$$

where the coefficients A_2, A_3 $\cdots$ A_7 $\cdots$ are solutions to (9), numerical integrators with symmetric increment functions have asymptotic expansions of the form:

$$u(t_{q+\triangle t}) = y(t_{q+\triangle t}) + A_2 \triangle t^2 + A_4 \triangle t^4 + A_6 \triangle t^6 + \cdots \tag{13}$$

Examples of `implicit` numerical methods having symmetric increment functions are the midpoint rule

$$y(t_{q+h}) = y(t_q) + h \cdot f(\frac{t_q + t_{q+h}}{2}, \frac{y(t_q) + y(t_{q+h})}{2}) \tag{14}$$

and the trapezoidal method

$$y(t_{q+h}) = y(t_q) + \frac{h}{2} \cdot [f(t_q, y(t_q)) + f(t_{q+h}, y(t_{q+h}))]. \tag{15}$$

Now let $\mathrm{u}^m_{n_k}(y(t_q), \triangle t)$ be the computed numerical solution from n_k steps of length $\triangle t/n_k$ using a numerical method of m^{th} order accuracy. Furthermore, assume that the computation starts from exact solution $y(t_q)$. Richardson observed [20,21] that improved numerical approximations could be obtained by solving the same problem over a prescribed base step $\triangle t$ several times with successively smaller internal timestep lengths. If the underlying numerical methods is 2nd order accurate, then this gives a sequence of numerical approximations:

$$\mathrm{u}^2_{n_1}(y(t_q), \triangle t), \mathrm{u}^2_{n_2}(y(t_q), \triangle t), \cdots, \mathrm{u}^2_{n_k}(y(t_q), \triangle t)$$

for the set of increasing integers $n_1, n_2 \cdots n_k$. Several numerical sequences have been proposed, including that of Romberg $\{1, 2, 4, 8, 16, 32, \cdots\}$, Bulirsch [3] $\{1, 2, 4, 6, 8, 12, \cdots\}$, and the double harmonic sequence $\{2, 4, 6, 8, 10, 12, 14, \cdots\}$. In each case, linear combinations of the numerical approximations are taken to eliminate the coefficients A_i. Successively refining $\triangle t$ according to the Romberg sequence, for example, gives:

$$\mathrm{u}^2_1(y(t_q), \triangle t) = y(t_{q+\triangle t}) + \sum_{i=1}^{i=\infty} A_{2i}[\triangle t]^{2i} \tag{16}$$

$$\mathrm{u}^2_2(y(t_q), \triangle t) = y(t_{q+\triangle t}) + \sum_{i=1}^{i=\infty} A_{2i}\left[\frac{\triangle t}{2}\right]^{2i} \tag{17}$$

Now the benefits of (13) compared to (12) are evident - subtracting (16) from 4 times equation (17) gives:

$$\begin{aligned}[4 \cdot y(t_{q+\triangle t}) - y(t_{q+\triangle t})] &= \left[4 \cdot \mathrm{u}^2_2(\cdot) - \mathrm{u}^2_1(\cdot)\right] + \mathrm{O}\left[\triangle t^4\right] \\ &= [4-1] \cdot \mathrm{u}^4_2(\cdot) + \mathrm{O}\left[\triangle t^4\right] \end{aligned} \tag{18}$$

thereby estimating the integral with two additional order of accuracy. For the Romberg sequence of internal timestep lengths it is relatively straight forward to show that successive

applications of the formula:

$$\mathrm{u}_{(2r)}^{(s+2)}(\cdot) = \left[\frac{4^{(s/2)}\mathrm{u}_{(2r)}^{(s)}(\cdot) - \mathrm{u}_{(r)}^{(s)}(\cdot)}{4^{(s/2)} - 1}\right] \tag{19}$$

will systematically eliminate higher order coefficients in the error polynomial. Here r is the number of intervals taken to integrate across the Δt time-step, and s the order of accuracy for the numerical estimate.

5 NEWMARK FAMILY OF INTEGRATION METHODS

Newmark integration methods [12,18] approximate the time dependent response of linear and nonlinear 2nd-order equations by insisting that equilibrium be satisfied only at a discrete number of points (or timesteps). If t_q and $t_{q+\Delta t}$ are successive timesteps in the integration procedure, then the two equations of equilibrium that must be satisfied are:

$$M\left[\dot{x}(t_q), x(t_q)\right]\ddot{x}(t_q) + F\left[\dot{x}(t_q), x(t_q)\right] = P\left(t_q\right) \tag{20}$$

and

$$M\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right]\ddot{x}(t_{q+\Delta t}) + F\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right] = P\left(t_{q+\Delta t}\right). \tag{21}$$

Now let's assume that solutions to (20) are known, and (21) needs to be solved. At each timestep there are $3n$ unknowns corresponding to the displacement, velocity, and acceleration of each component of x. Since we only have n equations, the natural relationship existing between the acceleration and velocity, and velocity and displacement must be enforced to reduce the number of unknowns to n. That is:

$$\dot{x}(t_{q+\Delta t}) = \dot{x}(t_q) + \int_{t_q}^{t_{q+\Delta t}} \ddot{x}(\tau)d\tau \tag{22}$$

$$x(t_{q+\Delta t}) = x(t_q) + \int_{t_q}^{t_{q+\Delta t}} \dot{x}(\tau)d\tau \tag{23}$$

where $\dot{x}(t_q)$ and $x(t_q)$ are the velocity and displacement at timestep t_q, and $\ddot{x}(\tau)$ is an unknown function.

The Newmark family of integration methods assume that: (a) acceleration within the timestep behaves in a prescribed manner, and (b) the integral of acceleration across the timestep may be expressed as a linear combination of accelerations at the endpoints. Discrete counterparts to (22)-(23) for the update velocity and displacement are:

$$\dot{x}(t_{q+\triangle t}) = \dot{x}(t_q) + \triangle t[(1-\gamma)\ddot{x}(t_q) + \gamma\ddot{x}(t_{q+\triangle t})] \tag{24}$$

$$x(t_{q+\triangle t}) = x(t_q) + \triangle t\dot{x}(t_q) + \frac{\triangle t^2}{2}[(1-2\beta)\ddot{x}(t_q) + 2\beta\ddot{x}(t_{q+\triangle t})] \tag{25}$$

with the parameters γ and β determining the accuracy and stability of the method under consideration. Equations (24) and (25) are substituted into (21), written in the form

$$g(\cdot) = M\left[\ddot{x}(t_{q+\triangle t})\right] \cdot \ddot{x}(t_{q+\triangle t}) \; - \; P(t_{q+\triangle t}) \; = \; 0$$

and solved by iteration. Finally, $\ddot{x}(t_{q+\triangle t})$ is back substituted into (24) and (25) for the update in velocity and displacement.

Remark 1 : It is well known that when $\gamma = 1/2$ and $\beta = 1/4$, acceleration is constant within the timestep $t \in [t_q, t_{q+\triangle t}]$, and equal to the average of the endpoint accelerations, i.e:

$$\ddot{x}(t) = \left[\frac{\ddot{x}(t_q) + \ddot{x}(t_{q+\triangle t})}{2}\right] = \ddot{x}(t_q) + \left[\frac{\ddot{x}(t_{q+\triangle t}) - \ddot{x}(t_q)}{2}\right] = \ddot{x}(t_q) + \left[\frac{\triangle\ddot{x}(t_q)}{2}\right] \tag{26}$$

In such cases, approximations to the velocity and displacement will be linear and parabolic, respectively. Moreover, when (20) takes the form:

$$[m]\,\ddot{x}(t_q) + [c]\,\dot{x}(t_q) + [k]\,x(t_q) = P(t_q) \tag{27}$$

equations (21) may be written

$$[M]\,\triangle\ddot{x}(t_q) = [\triangle P] \tag{28}$$

$$\text{where} \quad [M] = [m] + \frac{\triangle t}{2}[c] + \frac{\triangle t^2}{4}[k]$$

$$\text{and} \quad [\triangle P] = P(t_{q+\triangle t}) - P(t_q) - \triangle t\,[c]\,\ddot{x}(t_q) - \triangle t\,[k]\left[\dot{x}(t_q) + \frac{\triangle t}{2}\ddot{x}(t_q)\right].$$

This discrete approximation is second order accurate and unconditionally stable. It conserves energy exactly for the free response vibration of linear undamped SDOF systems; see Chapter 9, page 512 of Hughes [12] for a proof. This property will be important for the algorithm presented in the following sections.

6 NEWMARK-EXTRAPOLATION ALGORITHM

The algorithm proposed in this paper draws on ideas from Newmark integration and Richardson's Extrapolation. The approach is motivated by a desire to work directly with 2nd order equations, if possible, and by the observation that improved numerical estimates of response will result when equations (22) and (23) are evaluated with increased precision.

Instead of simply refining the step length of the numerical method, the idea is to integrate across the time interval $t \in [t_q, t_{q+\Delta t}]$ several times with successively refined step lengths. In each case, the Newmark method is used as the base integrator. It is proposed that with a judicious choice of Newmark parameters γ and β the underlying method will be second order accurate, stable, and have asymptotic error expansions for the velocity and displacement components containing only even ordered terms. Thus, extrapolation may be used to obtain improved estimates of displacement and velocity, with updates in acceleration being computed via back substitution of the extrapolated displacements and velocities into the equations of equilibrium.

The step by step algorithm for P levels of extrapolation using a Newmark base method, parameter settings β and γ, and a Romberg sequence of refinement is:

[1a] **Initialization :** For each component of velocity and displacement, dynamically allocate memory for a $(P \times P)$ extrapolation table. Lambert [16] reports that in practice P typically falls into the range 4-7.

[1b] Select a timestep Δt. In some cases this may be the maximum timestep length for algorithm stability, while in other instances it may be a suitably small Δt needed to draw a smooth graph of response.

[2a] **Outer Loop :** For Newmark Integration, q = 1 to nsteps

[2b] Set $t_q = [q-1] \cdot \triangle t$

[3a] **Inner Loop 1 :** For i = 1 to P

[3b] Incremental timestep $\triangle t_i = \left[\frac{\triangle t}{2^{(i-1)}}\right]$

[4a] **Inner Loop 2 :** For k = 1 to $2^{(i-1)}$

[4b] Calculate $P(t_k)$ at time $t_k = t_q + \mathrm{k} \cdot \triangle t_i$

[4c] Solve $M\left[\ddot{x}(t_k)\right] \cdot \ddot{x}(t_k) \;=\; P(t_k)$

[4d] Update : $\dot{x}(t_k) = \dot{x}(t_{k-1}) + \triangle t_i \cdot [(1-\gamma)\ddot{x}(t_{k-1}) + \gamma \ddot{x}(t_k)]$

$$x(t_k) = x(t_{k-1}) + \triangle t_i \cdot \dot{x}(t_{k-1}) + \left[\frac{\triangle t_i^2}{2}\right] \cdot [(1-2\beta)\ddot{x}(t_{k-1}) + 2\beta \ddot{x}(t_k)]$$

[4e] **End Inner Loop 2**

[3d] Put displacement $x(t_{q+\triangle t})$ and velocity $\dot{x}(t_{q+\triangle t})$ components in position $u_j^2(\cdot)$ of Extrapolation Table [1].

Steps	Stepsize	$\mathrm{O}\left[\triangle t^2\right]$	$\mathrm{O}\left[\triangle t^4\right]$	$\mathrm{O}[\cdots]$	$\mathrm{O}\left[\triangle t^{2^{(P-1)}}\right]$
1	$\triangle t$	$\mathrm{u}_1^2(\cdot)$			
2	$\triangle t/2$	$\mathrm{u}_2^2(\cdot)$	$\mathrm{u}_2^4(\cdot)$		
...	...	...	...	...	
$2^{(P-1)}$	$\triangle t/2^{(P-1)}$	$\mathrm{u}_{2^{(P-1)}}^2(\cdot)$	$\mathrm{u}_{2^{(P-1)}}^4(\cdot)$	...	$\mathrm{u}_{2^{(P-1)}}^{2(P+1)}(\cdot)$

Table [1] : Extrapolation Tableau for Romberg Sequence

[2c] **End Inner Loop 1 : Note :** Each component of this loop may be computed in parallel.

[1c] Use equation (19) to calculate columns 2-P of each extrapolation table. The lower

most right entries of Table [1], $u_{2(P-1)}^{2(P+1)}(\cdot)$, are taken as the starting displacement and velocity components for the next time-step.

[1d] Back substitute the displacement and velocity vectors from Step [1c] into equation (21), and solve for acceleration vector $\ddot{x}(t_{q+\triangle t})$.

[1e] **End Outer Loop**

The Newmark parameters γ and β are selected according to a dual set of criteria; in addition to requiring the base method be second order accurate and stable, asymptotic error expansions for updates in velocity and displacement must contain only even ordered terms.

6.1 Accuracy and Stability

For linear dynamical systems of the type mentioned above, the Newmark method is second order accurate if and only if $\gamma = 1/2$. Setting $\gamma = 1/2$ and $\beta = 1/4$ results a method that is unconditionally stable. For nonlinear systems $\dot{y} = f(t, y)$, which are twice continuously differentiable in t and y, the trapezoidal rule is stable, convergent, and second order accurate [13]. The other common choice of parameters is $\gamma = 1/2$ and $\beta = 1/6$; it corresponds to a linear time variation in acceleration, quadratic variation in velocity, and cubic variation in displacement. The price for increased accuracy (i.e. a smaller coefficient for the local truncation error) is conditional stability. This feature is of little benefit because the purpose of the extrapolation is to eliminate the truncation errors completely, irrespective of their size.

6.2 Asymptotic Error Expansion

The goal of this section is to determine which settings of γ and β result in asymptotic expansions for the local error containing only even ordered powers of Δt. First, assume that $M[\cdot]$ is invertible. Equations (20)-(21) may be written:

$$\ddot{x}(t_q) = M^{-1}\left[\dot{x}(t_q), x(t_q)\right] \cdot \left[P(t_q) - F\left[\dot{x}(t_q), x(t_q)\right]\right] \tag{29}$$

and

$$\ddot{x}(t_{q+\Delta t}) = M^{-1}\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right] \cdot \left[P(t_{q+\Delta t}) - F\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right]\right] \tag{30}$$

Substituting equations (29)-(30) into (24)-(25) and rearranging terms gives a pair of implicit equations for the update in velocity and displacement. They are:

$$\dot{x}(t_{q+\Delta t}) = \dot{x}(t_q) + \Delta t \cdot \Psi_1\left(t_q, t_{q+\Delta t}, x(t_q), \dot{x}(t_q), x(t_{q+\Delta t}), \dot{x}(t_{q+\Delta t}), \Delta t\right) \tag{31}$$

$$\begin{aligned}\text{where } \Psi_1(\cdot) = {} & (1-\gamma) \cdot M^{-1}\left[\dot{x}(t_q), x(t_q)\right] \cdot \left[P(t_q) - F\left[\dot{x}(t_q), x(t_q)\right]\right] + \\ & \gamma \cdot M^{-1}\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right] \cdot \left[P(t_{q+\Delta t}) - F\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right]\right]\end{aligned}$$

and

$$x(t_{q+\Delta t}) = x(t_q) + \Delta t \cdot \Psi_2\left(t_q, t_{q+\Delta t}, x(t_q), \dot{x}(t_q), x(t_{q+\Delta t}), \dot{x}(t_{q+\Delta t}), \Delta t\right) \tag{32}$$

$$\begin{aligned}\text{where } \Psi_2(\cdot) = {} & \dot{x}(t_q) + \left[\frac{\Delta t}{2}\right] \cdot (1-2\beta) \cdot M^{-1}\left[\dot{x}(t_q), x(t_q)\right] \cdot \left[P(t_q) - F\left[\dot{x}(t_q), x(t_q)\right]\right] + \\ & \left[\frac{\Delta t}{2}\right] \cdot 2\beta \cdot M^{-1}\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right] \cdot \left[P(t_{q+\Delta t}) - F\left[\dot{x}(t_{q+\Delta t}), x(t_{q+\Delta t})\right]\right].\end{aligned}$$

Assume that $\Psi_1(\cdot)$ and $\Psi_2(\cdot)$ are continuously differentiable with respect to the displacements and velocities over the complete time interval $[t_q, t_{q+\Delta t}]$. Equations (31) and (32)

will have even ordered asymptotic expansions if the interchange of parameters $t_q \leftrightarrow t_{q+\triangle t}$, $x(t_q) \leftrightarrow x(t_{q+\triangle t})$, $\dot{x}(t_q) \leftrightarrow \dot{x}(t_{q+\triangle t})$ and $\triangle t \leftrightarrow -\triangle t$ leaves the equations invariant. In other words, equation (31) must be identical to

$$\dot{x}(t_q) = \dot{x}(t_{q+\triangle t}) - \triangle t \cdot \Psi_1 \left(t_{q+\triangle t}, t_q, x(t_{q+\triangle t}), \dot{x}(t_{q+\triangle t}), x(t_q), \dot{x}(t_q), , -\triangle t\right) \quad (33)$$

where $\Psi_1(\cdot) = (1-\gamma) \cdot M^{-1}\left[\dot{x}(t_{q+\triangle t}), x(t_{q+\triangle t})\right] \cdot \left[P\left(t_{q+\triangle t}\right) - F\left[\dot{x}(t_{q+\triangle t}), x(t_{q+\triangle t})\right]\right] +$

$$\gamma \cdot M^{-1}\left[\dot{x}(t_q), x(t_q)\right] \cdot \left[P\left(t_q\right) - F\left[\dot{x}(t_q), x(t_q)\right]\right]$$

Equating coefficients gives $1-\gamma = \gamma$, i.e. $\gamma = 1/2$. Similarly, the displacements will have even ordered asymptotic expansions if the interchange of parameters $t_q \leftrightarrow t_{q+\triangle t}$, $x(t_q) \leftrightarrow x(t_{q+\triangle t})$, $\dot{x}(t_q) \leftrightarrow \dot{x}(t_{q+\triangle t})$ and $\triangle t \leftrightarrow -\triangle t$ in

$$x(t_{q+\triangle t}) = x(t_q) + \triangle t \dot{x}(t_q) + \frac{\triangle t^2}{2}[(1-2\beta)\ddot{x}(t_q) + 2\beta\ddot{x}(t_{q+\triangle t})] \quad (34)$$

gives a numerical scheme equivalent to

$$x(t_q) = x(t_{q+\triangle t}) - \triangle t \dot{x}(t_{q+\triangle t}) + \frac{\triangle t^2}{2}[(1-2\beta)\ddot{x}(t_{q+\triangle t}) + 2\beta\ddot{x}(t_q)] \quad (35)$$

Recall that $\gamma = 1/2$. Substituting

$$\dot{x}(t_{q+\triangle t}) = \dot{x}(t_q) \; + \; \frac{\triangle t}{2} \cdot [\ddot{x}(t_q) \; + \; \ddot{x}(t_{q+\triangle t})]$$

into (35) and rearranging terms gives

$$\begin{aligned} x(t_{q+\triangle t}) =& x(t_q) + \triangle t \dot{x}(t_q) + \\ & \frac{\triangle t^2}{2}[\ddot{x}(t_q) + \ddot{x}(t_{q+\triangle t})] - \frac{\triangle t^2}{2}[(1-2\beta)\ddot{x}(t_{q+\triangle t}) + 2\beta\ddot{x}(t_q)]. \end{aligned} \quad (36)$$

It is evident that the coefficients in (34) and (36) will be identical for any value of β.

Remark 2: A natural question to ask is whether or not "other combinations of γ and β are admissible when the system dynamics are linear ?" Consider, for example, the linear SDOF model

$$\ddot{x}(t_q) + 2\varepsilon\omega\dot{x}(t_q) + \omega^2 x(t_q) = P(t_q).$$

For the purposes of conducting the analysis, it is convenient to define the column vector $z(t_q) = [x(t_q), \dot{x}(t_q)]^T$ and write the discrete update for Newmark in first order form

$$A_1\,[\triangle t]\cdot z(t_{q+\triangle t}) = A_2\,[\triangle t]\cdot z(t_q) + L[t_q, t_{q+\triangle t}, \triangle t] \tag{37}$$

$$\text{where} \qquad A_1\,[\triangle t] = \begin{bmatrix} 1+\triangle t^2\beta\omega^2 & 2\triangle t^2\beta\gamma\omega \\ \triangle t\gamma\omega^2 & 1+2\varepsilon\triangle t\gamma\omega \end{bmatrix}$$

$$A_2\,[\triangle t] = \begin{bmatrix} 1-\triangle t^2/2(1-2\beta)\omega^2 & \triangle t(1-\triangle t(1-2\beta)\varepsilon\omega) \\ -\triangle t(1-\gamma)\omega^2 & 1-2\triangle t(1-\gamma)\varepsilon\omega \end{bmatrix}$$

$$\text{and} \quad L\,[t_q, t_{q+\triangle t}, \triangle t] = \begin{bmatrix} \triangle t^2/2\cdot[(1-2\beta)P(t_q) \;+\; 2\beta P(t_{q+\triangle t})] \\ \triangle t\cdot[(1-\gamma)P(t_q) \;+\; \gamma P(t_{q+\triangle t})] \end{bmatrix}.$$

Recasting equations (37) in the form of (6) gives

$$\dot{z}(t_{q+\triangle t}) = \dot{z}(t_q) + \triangle t\cdot\Psi_1\,(t_q, t_{q+\triangle t}, z(t_q), \triangle t) \tag{38}$$

$$\text{where} \quad \Psi_1(\cdot) = \left[\frac{A_1^{-1}[\triangle t]}{\triangle t}\right]\cdot[(A_2[\triangle t] \;-\; A_1[\triangle t])\cdot z(t_q) \;+\; L[t_q, t_{q+\triangle t}, \triangle t]]. \tag{39}$$

In is important to note that in (38) the notation $\Psi_1(\cdot)$ is used for a $(2n\times 1)$ column vector composed of equations $\Psi_1(\cdot)$ and $\Psi_2(\cdot)$ in (31) and (32), respectively. The interchange of

parameters $t_q \leftrightarrow t_{q+\triangle t}$, $\triangle t \leftrightarrow -\triangle t$, and $z(t_q) \leftrightarrow z(t_{q+\triangle t})$ will leave the numerical method unchanged if and only if

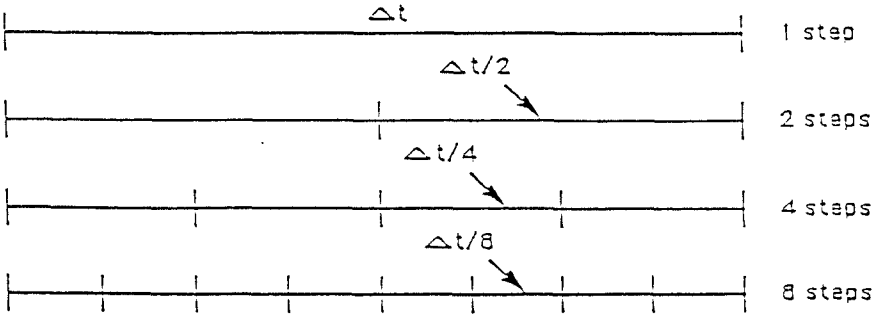
$$A_1^{-1}[\triangle t] \cdot A_2[\triangle t] = A_2^{-1}[-\triangle t] \cdot A_1[-\triangle t] \tag{40}$$

$$\text{and} \quad L[t_q, t_{q+\triangle t}, \triangle t] \; + \; A_1[\triangle t] \cdot A_2^{-1}[-\triangle t] \cdot L[t_{q+\triangle t}, t_q, -\triangle t] = 0. \tag{41}$$

Plugging equations (40)-(41) into `Mathematica` [27] and solving for solutions gives $\gamma = 1/2$ and any value β.

7 NUMERICAL EXPERIMENTS

This section presents the results of four numerical experiments; three examples and one counter example. Unless otherwise stated, simulations are conducted using $\gamma = 1/2$ and $\beta = 1/4$. The extrapolation computations consists of four levels of stepwise refinement matching the Romberg sequence, i.e. Δt, $\Delta t/2$, $\Delta t/4$, and $\Delta t/8$, i.e.:



They are conducted without stepsize and error control in the sense of Deuflhard [7]. In order to benchmark the accuracy of the proposed Newmark-Extrapolation algorithm against traditional approaches (for equivalent computational work per unit increment of time), the timestep for the standard Newmark integration procedure is divided by 15 - i.e. step length $\left[\frac{\Delta t}{15}\right]$.

The accumulation of differences between analytic and numerical response quantities is captured by defining the area of component error as:

$$\text{Error Area} = \text{stepsize} \cdot \sum_{i=1}^{i=N} |\text{Analytic}(t) - \text{Numerical}(t)|_i \tag{42}$$

where N equals the total number of integration timesteps over the simulation time interval. The parameters $Analytical(t)$ and $Numerical(t)$ are analytic and numerical solutions

Component x : Analytic x(0.03) = -0.47884882915567743984					
Steps	Stepsize	$O\left[\triangle t^2\right]$	$O\left[\triangle t^4\right]$	$O\left[\triangle t^6\right]$	$O\left[\triangle t^8\right]$
1	$\triangle t$	-0.47827819849			
2	$\frac{\triangle t}{2}$	-0.47870594155	-0.47884852258		
4	$\frac{\triangle t}{4}$	-0.47881309285	-0.47884880995	-0.47884882911	
8	$\frac{\triangle t}{8}$	-0.47883989418	-0.47884882795	-0.47884882915	-0.478848829155675

Component $\dot{x}$: Analytic $\dot{x}(0.03)$ = 0.99280863585386625224					
Steps	Stepsize	$O\left[\triangle t^2\right]$	$O\left[\triangle t^4\right]$	$O\left[\triangle t^6\right]$	$O\left[\triangle t^8\right]$
1	$\triangle t$	0.99282582702			
2	$\frac{\triangle t}{2}$	0.99281294252	0.9928086477		
4	$\frac{\triangle t}{4}$	0.99280971308	0.9928086366	0.99280863586	
8	$\frac{\triangle t}{8}$	0.99280890519	0.9928086359	0.99280863585	0.992808635853866

Table [2] : Simulation components after 1 Time Step

to components of response (acceleration, velocity, and displacement), and system invariants (total angular momentum, energy and so on). All calculations were done in `double precision` arithmetic on a SUN workstation.

7.1 Example 1 : Undamped SDOF Oscillator

The free vibration response of an undamped SDOF oscillator is studied as a means of gaining insight into the behavior and properties of the Newmark-extrapolation algorithm. If the mass m = 1 and stiffness k = 16, with initial conditions x(0) = 1 and $\dot{x}(0) = 0$, then the analytic solution is:

$$x(t) = \cos(4t) \tag{43}$$

Propagation of Energy Errors					
Steps	Stepsize	$\mathrm{O}\left[\triangle t^2\right]$	$\mathrm{O}\left[\triangle t^4\right]$	$\mathrm{O}\left[\triangle t^6\right]$	$\mathrm{O}\left[\triangle t^8\right]$
1	$\triangle t$	0.000000			
2	$\frac{\triangle t}{2}$	1.77e-15	4.12e-08		
4	$\frac{\triangle t}{4}$	1.77e-15	2.58e-09	1.11e-11	
8	$\frac{\triangle t}{8}$	-3.55e-15	1.61e-10	1.66e-13	-6.21e-15

Table [3] : Energy Errors for Newmark-Extrapolation at Time Step 1

The simulation timestep in the Newmark-Extrapolation algorithm was set at $\triangle t = 0.03$ seconds. For equivalent computational work per unit time, the simulations were repeated for the standard Newmark algorithm using $\triangle t = 0.002$ seconds and 15 times the number of time steps.

Performance of Numerical Algorithm at Step 1

Tables [2] summarizes the component response of the Newmark-Extrapolation algorithm after only 1 timestep; i.e. $t = 0.03$ seconds. The entries of column 3 correspond to velocity and displacement responses computed with the standard Newmark Method. Once column 3 is filled in, equation (19) is used for the calculation of columns 4-6, with elements $u_8^8(\cdot)$ being taken as the starting displacement and velocity at timestep 2. A key point to note from Table [2] is that convergence of the numerical solution to the analytic solution is much faster due to extrapolation (across the rows of the table) than by reduction of the step size (moving down the columns).

Physical considerations dictate that energy will be conserved in (43). Table [3] tracks the errors in the system energy at each stage of the extrapolation process for time step 1.

As indicated in the opening sections, the standard Newmark method with $\gamma = 1/2$ and $\beta = 1/4$ theoretically conserves energy exactly for this application, irrespective of the timestep length. In practice, however, minor errors are introduced due to numerical roundoff (see column 3 of Table [3]). Column 4 of Table [2] contains displacement and velocity estimates that are 4th order accurate. The corresponding column in Table [3] contains large errors in the energy of the numerical response; this observation is consistent with Dahlquist's stability criterion for this integration linear systems using a multi-step method [5]. As the extrapolation process continues, progressively higher order estimates of response quantities are obtained, and the energy error systematically approaches zero. Indeed, the final error in energy is of the same order as would occur due to numerical roundoff.

Long Term performance of Numerical Algorithm

The numerical response of the Newmark-Extrapolation algorithm was calculated for 3000 seconds, using to 100,000 time steps of $\Delta t = 0.03$ seconds. This corresponds to approximately 1910 full cycles of (43).

The first block of simulations corresponds to parameter settings $\gamma = 1/2$ and $\beta = 1/4$. The Newmark-Extrapolation algorithm (see column 3 of Table [4]) tracks the time variation in acceleration, velocity and displacement quantities much closer than the standard Newmark method. The error area in energy for the standard Newmark method should theoretically be zero. However, an accumulation in roundoff errors gives an absolute integral of energy errors 4.712×10^{-11}. The corresponding error area in energy given by Newmark-Extrapolation is of the same order as the displacements and velocity, as expected.

A second set of simulations were conducted using $\gamma = 1/2$ and $\beta = 1/6$. The objective

Error Component	Standard Newmark	Newmark Extrapolation
Displacement	4.074×10^{0}	5.304×10^{-8}
Velocity	1.629×10^{1}	2.121×10^{-7}
Acceleration	6.518×10^{1}	8.487×10^{-7}
Energy	4.712×10^{-11}	9.514×10^{-8}

Table [4] : Error Area after 3000 Seconds : gamma = 1/2 : beta = 1/4.

Error Component	Standard Newmark	Newmark Extrapolation
Displacement	2.037×10^{0}	8.474×10^{-9}
Velocity	8.148×10^{0}	3.389×10^{-8}
Acceleration	3.259×10^{1}	1.356×10^{-7}
Energy	4.267×10^{-3}	1.186×10^{-8}

Table [5] : Error Area after 3000 Seconds : gamma = 1/2 : beta = 1/6.

was to verify that extrapolation would work for values of β other than 1/4, and to see how much a linear variation in acceleration within the time step - versus constant acceleration - affects numerical accuracy. In the analysis of a linear undamped free vibrating SDOF systems, it can be shown - see Chapter 9 of Hughes [12] - that this Newmark method does not conserve energy exactly, and is stable only when $\omega \cdot \Delta t \leq 3.464$. For this application, the product $\omega \cdot \Delta t = 0.12$, so stability is not a problem.

The results of Tables [4] and [5] indicate that a linear approximation in acceleration improves the numerical accuracy in displacements, velocities, and accelerations in both the standard Newmark method and the proposed Newmark-Extrapolation algorithm. For example, estimates of displacement from Newmark-Extrapolation, have error areas

53.04×10^{-9} and 8.474×10^{-9} for the constant and linear acceleration approximations, respectively. This improvement in numerical accuracy is minor, compared to that obtained via extrapolation.

Comparison to Mid-point Rule

In the numerical solution of complicated dynamical systems, often the equations of motion are written

$$g(\cdot) = M\left[\ddot{x}(t_{q+\triangle t})\right] \cdot \ddot{x}(t_{q+\triangle t}) - P(t_{q+\triangle t}) = 0$$

and solved by iteration at each timestep. One problem with this approach is the potential destruction of the asymptotic error expansion if the iterative solution to the equations is not sufficiently accurate; that is, compared to the so-called `exact` method using LU decomposition. To see if this was likely to happen, the SDOF dynamics were written

$$\begin{bmatrix} \dot{p}(t) \\ \dot{x}(t) \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & k \end{bmatrix} \cdot \begin{bmatrix} p(t) \\ x(t) \end{bmatrix}. \tag{44}$$

Here $x(t)$ and $p(t)$ are the displacement and momentum - $p(t) = m \cdot \dot{x}(t)$ - of the mass-spring system. The discrete approximation to (44) using the midpoint rule is

$$\begin{bmatrix} \frac{p(t_{q+\triangle t})-p(t_q)}{\triangle t} \\ \frac{x(t_{q+\triangle t})-x(t_q)}{\triangle t} \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & k \end{bmatrix} \cdot \begin{bmatrix} \frac{p(t_{q+\triangle t})+p(t_q)}{2} \\ \frac{x(t_{q+\triangle t})+x(t_q)}{2} \end{bmatrix}. \tag{45}$$

The alert reader will notice that energy is conserved when equations (45) are solved exactly. Furthermore, in the solution of linear dynamical systems, the midpoint and trapezoidal rules are equivalent simply because:

$$\triangle t \cdot f\left(\frac{x(t_q) + x(t_{q+\triangle t})}{2}\right) = \frac{\triangle t}{2} \cdot \left[f\left(x(t_q)\right) + f\left(x(t_{q+\triangle t})\right)\right]$$

Error Component	Newmark-Extrapolation	Midpoint-Extrapolation
Displacement	5.304×10^{-8}	5.222×10^{-8}
Velocity	2.121×10^{-7}	2.089×10^{-7}
Energy	9.514×10^{-8}	9.498×10^{-8}

Table [6] : Error Area after 3000 Seconds

Numerical solutions to (45) were calculated for 100,000 timesteps of $\triangle t = 0.03$ seconds, the equations being solved at each timestep using the iterative procedure described in Section 7.3. Table [6] shows the error areas for the Newmark-Extrapolation and the midpoint formulation are similar, a good indication that the iterative equation solver is working well.

7.2 Example 2 : Damped SDOF subject to External Loading

Consider the damped SDOF subject to external loads:

$$\ddot{x}(t) + 4\dot{x}(t) + 13x(t) = \left[\frac{1}{3}\right] e^{-2t} \cdot \sin[3t] \tag{46}$$

If x(0) = 1 and $\dot{x}(0) = -2$, then Laplace transforms gives an analytic solution:

$$x(t) = e^{-2t} \cdot \cos[3t] + \left[\frac{e^{-2t}}{54}\right][\sin(3t) - 3t\cos(3t)]. \tag{47}$$

The damped natural period of this system is approximately 2.3 seconds.

Simulation Results

200 timesteps of simulation were computed using Newmark parameters $\gamma = 1/2$, $\beta = 1/4$, and timestep length $\triangle t = 0.03$ seconds. This corresponds to approximately 3 full

cycles of motion. Table [6] shows that the standard Newmark algorithm produces error areas of the order 10^{-6}. When the Newmark-Extrapolation algorithm is used to compute the response, the same error areas are of the order 10^{-15}.

Error Component	Standard Newmark	Newmark Extrapolation
Displacement	2.32×10^{-6}	8.067×10^{-16}
Velocity	9.30×10^{-6}	3.223×10^{-16}
Acceleration	3.36×10^{-6}	1.235×10^{-15}

Table [7] : Error Area after 200 Time Steps

7.3 Example 3 : Lagrangian Formulation for 2-Body Problem

The third application is a study of the dynamical behavior of two planar rigid bodies connected by a frictionless revolute joint in a zero-gravity environment. Let the bodies have masses m_1 and m_2, and moments of inertia about their centers of mass I_1 and I_2. Figure [1] shows that the centers of mass for each body are located at distances d_1 and d_2 from the revolute joint. Moreover, the orientation of each body is described by the angle - $\theta_1(t)$ and $\theta_2(t)$ - it makes to the x-axis, measured in an anti-clockwise direction.

In the absence of external torque the system is conservative. The total angular momentum of the system is

$$M = [1 \quad 1] \begin{bmatrix} \tilde{I}_1 & \epsilon \cdot d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t)) \\ \epsilon \cdot d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t)) & \tilde{I}_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1(t) \\ \dot{\theta}_2(t) \end{bmatrix}. \quad (48)$$

and the Lagrangian of the system

$$L = \frac{1}{2} \left[\tilde{I}_1 \dot{\theta}_1^2(t) + \tilde{I}_2 \dot{\theta}_2^2(t) + 2\dot{\theta}_1(t) \cdot \dot{\theta}_2(t) \epsilon \cdot d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t)) \right]. \quad (49)$$

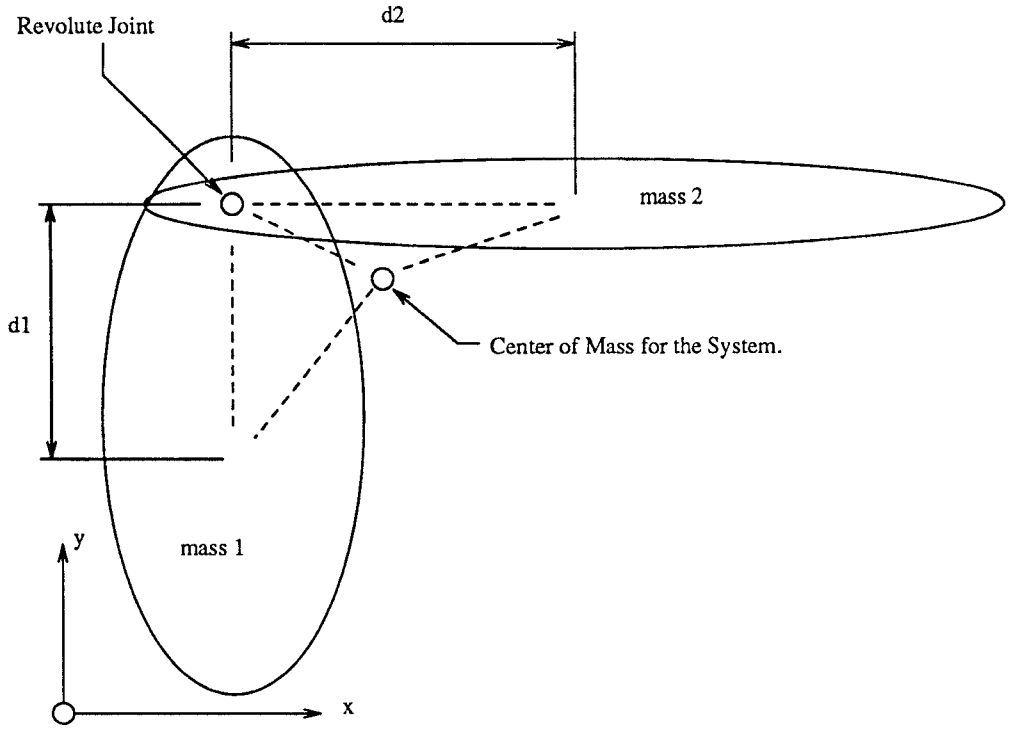


Figure [1] Configuration of Planar Two Body Dynamical System

In equations (48) and (49), $\epsilon = (m_1 m_2)/(m_1 + m_2)$ is a reduced mass, and $\tilde{I}_1 = I_1 + \epsilon d_1^2$ and $\tilde{I}_2 = I_2 + \epsilon d_2^2$ are augmented inertias of the bodies. The Euler-Lagrange equation is used to derive (see Sreenath et al. [23,24] for details) the equations of motion for this system. They are:

$$\begin{bmatrix} \tilde{I}_1 & \epsilon d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t)) \\ \epsilon d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t)) & \tilde{I}_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1(t) \\ \ddot{\theta}_2(t) \end{bmatrix} +$$

$$\epsilon d_1 d_2 \cdot \sin(\theta_2(t) - \theta_1(t)) \begin{bmatrix} -\dot{\theta}_2^2(t) \\ \dot{\theta}_1^2(t) \end{bmatrix} = 0. \tag{50}$$

Notice that (50) is linear in angular accelerations, and falls into the class of problems covered in (1). If angles and velocities are provided as initial conditions, then the starting angular accelerations may be recovered directly from (50).

Computational Implementation

The first step of the computational procedure is to write (50) in component form, namely:

$$g_1\Big(\ddot{\theta}_1(t), \ddot{\theta}_2(t)\Big) = \tilde{I}_1\ddot{\theta}_1(t) + \epsilon \cdot d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t))\ddot{\theta}_2(t) - \dot{\theta}_2^2(t) \cdot \epsilon d_1 d_2 \cdot \sin(\theta_2(t) - \theta_1(t)) = 0, \tag{51}$$

$$g_2\Big(\ddot{\theta}_1(t), \ddot{\theta}_2(t)\Big) = \tilde{I}_2\ddot{\theta}_2 + \epsilon \cdot d_1 d_2 \cdot \cos(\theta_2(t) - \theta_1(t))\ddot{\theta}_1(t) + \dot{\theta}_1^2(t) \cdot \epsilon \cdot d_1 d_2 \cdot \sin(\theta_2(t) - \theta_1(t)) = 0, \tag{52}$$

where the subscripts take there usual meaning. The number of unknown variables at each timestep is reduced to two by prescribing constant angular acceleration across each time interval $t \in [t_q, t_{q+\Delta t}]$. In an analogous manner to equations (22)-(23), it follows that:

$$\dot{\theta}(t_{q+\Delta t}) = \dot{\theta}(t_q) + \int_{t_q}^{t_{q+\Delta t}} \ddot{\theta}(\tau)d\tau = \dot{\theta}(t_q) + \frac{\Delta t}{2}\left[\ddot{\theta}(t_q) + \ddot{\theta}(t_{q+\Delta t})\right], \tag{53}$$

and

$$\theta(t_{q+\Delta t}) = \theta_q + \int_{t_q}^{t_{q+\Delta t}} \dot{\theta}(\tau)d\tau = \theta(t_q) + \dot{\theta}(t_q)\Delta t + \frac{\Delta t^2}{4}\left[\ddot{\theta}(t_q) + \ddot{\theta}(t_{q+\Delta t})\right]. \tag{54}$$

Substituting (53)-(54) into (51)-(52) gives two nonlinear equations where the only unknowns are $\ddot{\theta}(t_{q+\Delta t})$. An iterative solution is computed by letting:

$$\ddot{\theta}_{1,q+\Delta t}^{k+1} = \ddot{\theta}_{1,q+\Delta t}^{k} + \Delta\ddot{\theta}_1^k, \tag{55}$$

$$\ddot{\theta}_{2,q+\Delta t}^{k+1} = \ddot{\theta}_{2,q+\Delta t}^{k} + \Delta\ddot{\theta}_2^k. \tag{56}$$

where $\ddot{\theta}_{i,q+\Delta t}^{k}$ is the k^{th} iterate for acceleration component i at time $t_{q+\Delta t}$; here we have dropped the notation $\ddot{\theta}(t_{q+\Delta t})$ only to make the equations more compact. A sequence of

iterative solutions is defined by taking a Taylor series expansions about $g_1(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t})$ and $g_2(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t})$ and truncating higher order terms. This gives:

$$g_1(\ddot{\theta}^{k+1}_{1,q+\Delta t}, \ddot{\theta}^{k+1}_{2,q+\Delta t}) = g_1(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t}) + \frac{\partial g_1}{\partial \ddot{\theta}^k_{1,q+\Delta t}} \Delta\ddot{\theta}^k_1 + \frac{\partial g_1}{\partial \ddot{\theta}^k_{2,q+\Delta t}} \Delta\ddot{\theta}^k_2 + \mathbf{O}((\Delta\ddot{\theta}^k_1)^2, (\Delta\ddot{\theta}^k_2)^2) = 0, \tag{57}$$

$$g_2(\ddot{\theta}^{k+1}_{1,q+\Delta t}, \ddot{\theta}^{k+1}_{2,q+\Delta t}) = g_2(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t}) + \frac{\partial g_2}{\partial \ddot{\theta}^k_{1,q+\Delta t}} \Delta\ddot{\theta}^k_1 + \frac{\partial g_2}{\partial \ddot{\theta}^k_{2,q+\Delta t}} \Delta\ddot{\theta}^k_2 + \mathbf{O}((\Delta\ddot{\theta}^k_1)^2, (\Delta\ddot{\theta}^k_2)^2) = 0. \tag{58}$$

Truncating the nonlinear terms of $\Delta\ddot{\theta}$ in (57)-(58), and rewriting in matrix form gives

$$\begin{bmatrix} \frac{\partial g_1}{\partial \theta^k_{1,q+\Delta t}} & \frac{\partial g_1}{\partial \theta^k_{2,q+\Delta t}} \\ \frac{\partial g_2}{\partial \theta^k_{1,q+\Delta t}} & \frac{\partial g_2}{\partial \theta^k_{2,q+\Delta t}} \end{bmatrix} \begin{bmatrix} \Delta\ddot{\theta}^k_1 \\ \Delta\ddot{\theta}^k_2 \end{bmatrix} = - \begin{bmatrix} g_1(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t}) \\ g_2(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t}) \end{bmatrix}, \tag{59}$$

which is solved for the incremental update in components (55)-(56). A concise statement of (59) is **J.h = -g**. Iterations continue at each timestep until: (a) a preset maximum number of iterations is reached, or (b) all changes in angular acceleration components from (55)-(56) are less than a preset error value times the magnitude of the acceleration imbalance at the beginning of the iterations. Moreover, divergence of the iterates is avoided by ensuring $\|g(\ddot{\theta}^{k+1}_{1,q+\Delta t}, \ddot{\theta}^{k+1}_{2,q+\Delta t})\|_2$ is less than $\|g(\ddot{\theta}^k_{1,q+\Delta t}, \ddot{\theta}^k_{2,q+\Delta t})\|_2$. When this test fails, **h** is divided by powers of 2 until the inequality is satisfied.

Simulation Results

Consider the response of the 2-body having properties $m_1 = 1$, $m_2 = 2$, $d_1 = 1.0$, $d_2 = 1.5$, $I_1 = 1.0$, and $I_2 = 3.0$. If the initial displacement and velocity vectors are $[0, 1]$ and $[0.0, 5.0]$, respectively, then back substituting into equation (50) gives (approximate) start-

Error Component	Standard Newmark	Proposed Algo
Lagrangian	4.421×10^{-3}	9.309×10^{-10}
Momentum Norm	1.351×10^{-3}	2.293×10^{-10}

Table [8] : Error Area after 1000 Time Steps

ing angular acceleration components $[13.133, -1.5768]$. The norm of angular momentum is 25.20151152934070 and the Lagrangian 56.25.

The response of the 2-body was calculated for 1000 timesteps at $\Delta t = 0.03$ seconds using the Newmark-Extrapolation algorithm (i.e. $t \in [0, 30]$ seconds). The computation was repeated using the Standard Newmark algorithm for 15000 timesteps at $\Delta t = 0.03/15 = 0.002$ seconds. In both cases this interval of simulation corresponds to approximately 10 full cycles of the coupled rigid body components.

As indicated in equations (48) and (49), theoretical considerations dictate that the body momentum and energy will be constant. Consequently, error areas for momentum and energy were calculated for both simulation cases, and are shown in Table [8]. They indicate that the Newmark-Extrapolation is significantly more accurate than the standard Newmark algorithm.

8.4 Counter Example 4 : Bilinear Mass-Spring System

The counter example demonstrates how the extrapolation process will fail when the underlying equations of motion are not sufficiently differentiable, as required by Gragg's theorem. Consider, for example, the free vibration of a simple mass-spring SDOF oscillator

having mass m, and a bilinear spring restoring force

$$F(x) = \begin{cases} 10x, & \text{zone 1 if } \mid x \mid \leq 1.0; \\ 10 * (1 + p(x-1)), & \text{zone 2 for } \mid x \mid > 1.0. \end{cases}$$

where x is the displacement of the mass and p is a coefficient of strain hardening. If the mass has displacement $x(0)$ and velocity $\dot{x}(0)$ at time t = 0, then the analytic solution for free vibration is a piecewise series of solutions

$$x(t) = A\cos(w(x)t) + B\sin(w(x)t)$$

for displacements within each zone. The instantaneous circular frequency of the system is given by $w(x) = \sqrt{[k(x)/m]}$, where k(x) = $dF(x)/dx$ is the tangent stiffness. The coefficients A and B depend on the initial conditions as the system passes from one zone to another. More important, let $\Psi(\cdot)$ be the increment function - in the sense of equation (6) - for the numerical method when the trapezoidal rule is used to approximate the update in velocities and displacements. There does not exist a Lipschitz constant L such that

$$\mid \frac{\partial \Psi(t, x_1)}{\partial x} - \frac{\partial \Psi(t, x_2)}{\partial x} \mid \leq \; L \mid x_1 \; - x_2 \mid$$

for all x_1 and x_2 in the neighborhood of x = ± 1. From a physical viewpoint, this translates to a discontinuity in the third derivative in displacement at the boundary between zones 1 and 2. The necessary conditions for Gragg's theorem are not satisfied, and as a result, the extrapolation should fail as the boundary between zones 1 and 2 is crossed.

Simulation Results

For the numerical experiments, mass m was set to 1, strain hardening ratio p set to 0.5, and the initial displacement and velocity at t = 0 set to 2 and 0, respectively. The system

Component	Analytic	Numerical	Error
Displacement	1.079367284205931	1.079367284205930	0.000000000000001
Velocity	-4.835406755059004	-4.835406755059001	-0.000000000000003
Acceleration	-10.396836421029658	-10.396836421029651	-0.000000000000007
Energy	17.500000000000000	17.499999999999972	0.000000000000028

Table [9] : System Components at end of Timestep 9

Component	Analytic	Numerical	Error
Displacement	0.877842947098626	0.877954601994794	-0.000111654896169
Velocity	-5.227371560478852	-5.224170476570483	-0.003201083908370
Acceleration	-8.778429470986259	-8.779546019947945	0.001116548961686
Energy	17.50000000000000	17.499999999954522	0.000000000045478

Table [10] : System Components at end of Timestep 10

has constant energy = 17.5. If $\triangle t = 0.04$ seconds, then the Newmark-Extrapolation scheme crosses from `zone2` into `zone1` during the 10th timestep (starting at t = 0.40 seconds).

Table [9] shows that the analytic and numerical system parameters agree to approximately 15 decimal places at the end of timestep 9. During timestep 10, each simulation block starts in zone 2 and moves to zone 1. It is important to note that all sub-steps of standard Newmark integration conserve energy, including those that cross the boundary between zones. Still, the extrapolation of displacements and velocities fails, and is accompanied by a significant loss in accuracy for the system energy. The latter results are summarized in Table [10].

8 CONCLUSIONS

This paper has described the formulation of a new class of algorithms for integrating linear second order equations, and those containing smooth nonlinearities. These algorithms are based on ideas taken from standard Newmark integration methods, and extrapolation techniques. For the algorithm to work, the underlying Newmark method must be stable, second order accurate, and produce asymptotic error expansions for the response quantities containing only even ordered terms. It has been shown that second order accuracy and a desirable asymptotic expansion is achieved for Newmark parameters $\gamma = 1/2$, and any value of β. This means that β may selected on the basis of stability considerations alone. The numerical experiments indicate that the Newmark-Extrapolation algorithm is capable (in certain situations) of tracking response quantities in excess of 1,000,000 times more accurately than the standard Newmark method.

The current plan is to extend this work in two directions. First, algorithms of the type developed here lend themselves to concurrent computing. A prototype environment for distributed numerical computations is currently being developed. The algorithm's performance in this environment will be reported in future papers. Second, it is planned to extend the extrapolation technique to single rigid body and coupled rigid body applications. The challenge here is to try and find ways of doing extrapolation within the Lie Algebra of the application at hand. By doing this, the hope is that it may be possible to compute refined estimates of system response without destroying invariants, such as conservation of energy.

9 ACKNOWLEDGEMENTS

This work was supported in part by the AFSOR University Research Initiative Program under grants AFSOR-87-0073 and AFSOR-90-0105, an by NSF Research Initiation Grant NSF BCS 8907722. The writer would also like to acknowledge lively discussions with L.S. Wang, B. Frankpitt, J. Gillis and P.S. Krishnaprasad during the course of this work.

10 REFERENCES

[1] M.A. Austin, P.S. Krishnaprasad, and L.S. Wang, "On Symplectic and Almost Poisson Integration of Rigid Body Systems," Submitted to *Journal of Computational Physics.* Also published as Systems Research Center Technical Report TR 91-45, May, 1991.

[2] M.A. Austin, and B.K. Voon, "Development of a Distributed Computing Environment for Optimization-Based CAD of Structures," **Optimization of Structural Systems and Industrial Applications**, Computational Mechanics Publications, Elsevier Applied Science, pp. 643-654. (Also presented at *Conf. on Computer Aided Optimum Design of Structures*, Boston, USA, 25-27 June, 1991).

[3] R. Bulirsch, and J. Stoer, "Numerical Treatment of Ordinary Differential Equations by Extrapolation Methods," Num Math, Vol. 8, pp 1-13, 1966.

[4] R.H. Byrne, "Interactive Graphics and Dynamical Simulation in a Distributed Processing Environment," Masters Degree Thesis Report, Systems Research Center, University of Maryland, College Park, MD 20742, 1990.

[5] G. Dahlquist, "A Special Stability Problem for Linear Multistep Methods," BIT 3, pp. 27-43, 1963.

[6] F. De Camarmond, and M. Steiner, "Progress on the Dynamical Simulation of Rigid Bodies," *Unpublished Report*, Systems Research Center, 1987.

[7] P. Deuflhard, "Order and Stepsize Control in Extrapolation Methods," Num. Math., Vol. 41, pp 399-422, 1983.

[8] W.B. Gragg, "On Extrapolation Algorithms for Ordinary Initial Value Problems," *Journal SIAM Numerical Analysis*, Vol, 2, No. 3, 1965.

[9] E. Hairer, S.P. Norsett, and G. Wanner, **Solving Ordinary Differential Equations 1 : Nonstiff Problems**, Springer-Verlag, 1987.

[10] E. Hairer, and C. Lubich, "Asymptotic Expansions of the Global Error for Fixed-Stepsize Methods," Numer. Math., Vol. 45. pp 345-360, 1984.

[11] C. Hoff, and R.L. Taylor, "Higher Derivative Explicit One Step Methods for Nonlinear Dynamic Problems. Part 11: Practical Calculations and Comparisons with Higher Order Methods," *International Journal for Numerical Methods in Engineering*, Vol. 29, pp. 291-301, (1990).

[12] T.R.J. Hughes, **The Finite Element Method : Linear Static and Dynamic Finite Element Analysis**, Prentice-Hall Inc, 1987.

[13] T.R.J. Hughes, "Stability, Convergence and Growth and Decay of Energy of the Average Acceleration Method in Nonlinear Structural Dynamics," *Computers and Structures*, Vol. 6, pp 313-324, 1976.

[14] F. Kang, "Symplectic Geometry and Numerical Methods in Fluid Dynamics," Springer Lecture Notes in Physics 264, *10th Int. Conf. Num. Methods in Fluid Mechanics*, Proc. Beijing, 1986.

[15] P.S. Krishnaprasad, "Lie-Poisson Structure, Dualspin Spacecraft and Asymptotic Stability," *Nonlinear Analysis, Theory, Methods and Applications*, Vol. 9, No. 10, pp. 1011-1035, 1985.

[16] J.D. Lambert, **Computational Methods in Ordinary Differential Equations**, Wiley and Sons, 1973.

[17] J. Marsden, O.M. O'Reilly, F. Wiklin, and B. Zombro, "Symmetry, Stability, Geometric Phases, and Mechanical Integrators," Department of Applied Mathematics, Cornell University, 1990. Preprint.

[18] N.M. Newmark, "A Method for Computation of Structural Dynamics," Proceedings of American Society of Civil Engineers, 85 EM3 (1959) pp 67-94.

[19] T. Posbergh, "Modeling and Control of Mixed and Flexible Structures," *Technical Report SRC TR 88-58*, Systems Research Center, University of Maryland, College Park, MD 20742.

[20] L.F. Richardson, "The Approximate Arithmetical solution by finite differences of Physical Problems including Differential Equations, with an Application to the Stresses in Masonry Dam," *Phil. Trans.*, A, Vol. 210, 1910, pp 307-357.

[21] L.F. Richardson, "The Deferred Approach to the Limit," *Phil. Trans.*, A, Vol. 226, 1927, pp 299-349.

[22] J. Simo, and K.K. Wong, "Unconditionally Stable Algorithms for Rigid Body Dynamics that Exactly Preserve Energy and Momentum," *International Journal of Numerical Methods in Engineering*, Vol. 31, pp. 19-52, (1991).

[23] N. Sreenath, Y.G. Oh, P.S. Krishnaprasad, and J. Marsden, "The Dynamics of Coupled Planar Rigid Bodies," *Technical Report SRC TR 86-56*, Systems Research Center, University of Maryland, College Park, MD 20742.

[24] N. Sreenath, "Modeling and Control of Multibody Systems," *Technical Report SRC TR 87-163*, Systems Research Center, University of Maryland, College Park, MD 20742.

[25] H.J. Stetter, "Asymptotic Expansions for the Error of Discretization Algorithms for Non-linear Functional Equations," *Numerical Mathematik*, Vol. 7, pp. 18-31, 1965.

[26] R. Yang, and P.S. Krishnaprasad, "On the dynamics of floating four-bar linkages", *Proceedings of 28th IEEE Conference on Decision and Control*, Vol. 2, pp. 1632-1635, 1989.

[27] S. Wolfram, **Mathematica : A system for Doing Mathematics by Computer**, Addison-Wesley Publishing Company, 1991.